

# Open Source Software and Economic Growth: A Classical Division of Labor Perspective

Giampaolo Garzarelli and Yasmina Reem Limam and Bjørn Thomassen

School of Economic and Business Sciences, University of the Witwatersrand

22. June 2007

# Open Source Software and Economic Growth: A Classical Division of Labor Perspective

Giampaolo Garzarelli (Corresponding author)
School of Economic and Business Sciences
University of the Witwatersrand
Private Bag X3, WITS 2050
Johannesburg, Republic of South Africa
**Giampaolo.Garzarelli@wits.ac.za**
Tel: +27.11.717-8128
Fax: +27.11.717-8081

Yasmina Reem Limam
Département de Sciences Economiques et de Méthodes Quantitatives
Faculté de Sciences Economiques et de Gestion de Nabeul
Route Hammamet El Mrezga
Nabeul 8000, Tunisia
**yrlimam@yahoo.com**

Bjørn Thomassen
Department of International Relations
The American University of Rome
Via Pietro Roselli 4
Rome 00153, Italy
**b.thomassen@aur.edu**

Forthcoming *Information Technology for Development*

This draft: June 22, 2007

# Open Source Software and Economic Growth: A Classical Division of Labor Perspective


## Abstract

The article turns to classical economic insights on the division of labor and to institutional reasoning to identify some costs and benefits of Open Source Software (OSS) and proprietary software production. It suggests that, thanks to its licenses, OSS favors market expansion more than proprietary software does by tapping into spontaneous work input. The spontaneous tapping leads to a division of labor that exhibits what the article calls redundant economies. By generating a circle of knowledge growth, reuse, and sharing, redundant economies lead to increasing returns, which are crucial for economic growth. (92 words.)



## JEL Codes

D20, L17, L23, O33, O34

## Keywords

Division of Labor, Extent of the Market, Increasing Returns, Institutions, Knowledge, Open Source Software, Redundant Economies.

# 1. Introduction

This article argues that just as computer technology can be an engine of economic growth so too can software.[1] The primary software it considers is not proprietary (e.g., Adobe, Microsoft), but Open Source (e.g., Apache, Linux).[2] The distilled essence of Open Source Software (OSS) is that it is a form of collaborative, volunteer software production that relies on shared code and open standards. Though basically present since the birth of the computer, the Internet has allowed OSS production to become more diffused thanks to the ability to upload and download information (including entire programs) and to communicate in real time.[3]

To make our argument we rely on classical economic insights. We offer some stylized facts about why OSS seems more promising than proprietary software for economic growth from a Smithian division of labor perspective (Smith [1776]1981). We attempt to do so by not forgetting the complement of Smithian division of labor theory, namely, the extent of the market. We basically agree with Young ([1929]1990, p. 161) that the

> vast and intricate system of economic organization, by means of which the varied needs of modern life are met, is mostly a product of a continuous process of evolution. In only a very small part is it a result of conscious collective planning or devising. It grows and changes unceasingly … . Every innovation, whether in the technique of production or in … organization … affects in some degree the conditions which govern the activities of other producers. The economic system grows and evolves, like a living organism, by means

[1] By economic growth we have in mind growth in real output per capita, also known as intensive growth.

[2] The **$100 Laptop** prototype unveiled at the **Summit on the Information Society** in Tunis in November 2005 runs, incidentally, on Linux.

[3] See, *inter alia*, Raymond (2001), Benkler (2002), Lerner and Tirole (2002), Lakhani and von Hippel (2003), UNCTAD (2003, Ch. 4), Dalle and David (2004), and Garzarelli (2004).

> of successive adjustments and adaptations. But change breeds change, and every new adjustment paves the way for another.

Or, in more fitting terms, we suggest that OSS is more promising because it favors market expansion more than proprietary software does. It is able to do so because OSS licenses enable tapping into spontaneous work input. In light of the porosity ('openness') of OSS organization of work – and, more generally, of the OSS community – such tapping generates a circle of knowledge growth and sharing that moves beyond the narrow confines of any unique software project. In other words, we essentially claim that the net value of the knowledge spillovers emanating from the OSS production mode is positive; and in order to make such claim we attempt to identify what is unique about OSS division of labor.

As elaborated in the next section, the backdrop of our story is institutional. That is, our division of labor story rests on the assumption that the rules of the game are not a matter of indifference: we stress that it is the nature of OSS licenses that enables an open division of labor where the benefits of redundant task matching outweigh the costs. We try to make our case by looking into the costs and benefits of proprietary and OSS organization of production. This is in line with our institutional backdrop. Institutional analysis is a comparative exercise: one compares the economic properties of alternative, feasible institutional and organizational forms; one does not compare the economic properties of organizations and institutions using an optimality measure, such as the Pareto one. A major point that emerges from our comparative exercise is that both proprietary and OSS organizations take advantage of the division of labor. But the division of labor benefits are not coextensive for the two organizations. The reason for this is that the two organizations rely on software licenses that engender different extents of the market.

In our suggested thesis the market expands not just for what we may think of as physical reasons, say for "the operations of specialized undertakings which, taken together, constitute a new industry" (Young 1928, p. 539). The market also expands because of the emergence of specialized knowledge that concurrently defines – and coevolves with – physical market expansion (Rosenberg 1963; Bresnahan and

Gambardella 1998). The OSS community in fact represents a particularly clear example of a learning community where the emergent knowledge has significant productive value. In effect, the OSS community is mainly about the exchange, production, and reuse of one 'service': knowledge (Garzarelli 2004).

Before proceeding, we wish to present our two primary motivations for relying on classical economic insights. Introducing our motivations will then also ease the more complete illustration of our implied analytical method.

## 2. Motivations, Method, and Economic Growth

A first motivation for which our *Weltanshauung* is classical is institutional. Institutions have repercussions. Institutions are an ensemble of rules that can create incentives to contribute to the creation of value (e.g., incentives to organize, to innovate, to bear risk), and in this way influence economy-wide division of labor arrangements (Foss and Garzarelli forthcoming). And different division of labor arrangements entail different increasing returns, which are a fundamental determinant of economic growth. Increasing returns, which generally emanate from increases in knowledge, in fact imply that output increases by a factor greater than the corresponding increase in input. For example, if there's knowledge reuse – as is often the case in software code production (Haefliger et al. forthcoming) – output can, for example, be doubled without doubling an input (knowledge) (Langlois 1999).

This view that sees increasing returns and institutions as inseparable is central to Smithian thought, but enters contemporary economic theories of growth only in an *ad hoc* fashion. For instance, the endogenous theory of growth (e.g., Romer 1986; Lucas 1988) is not in contradiction with the view that increasing returns are fundamental for economic growth because they are generated from the growth of knowledge that leads to innovation. But there is an important difference. In the endogenous theory the growth of knowledge mainly emanates from private research and development investments. Since the causality of the theory has it that growth is a direct consequence of R&D it follows that phenomena that reduce R&D – such as

knowledge spillovers – are not desirable for they reduce social welfare. In our view, however, this is not the case, because knowledge growth is not unique to private R&D. The benefits of learning are social in the sense that they occur in the process of their emergence (Langlois and Robertson 1996).[4]

The substantive point is that if we consider a scenario where "there are no increasing returns to institutions and markets are competitive, institutions do not matter" (North 1990, p. 95). But institutions matter because it is their nonhomogenous nature that determines different division of labor arrangements. And it is different divisions of labor that can generate the unequal degrees of social learning that we interpret to be a crucial ingredient to understand different growth trajectories. In different terms, especially in an environment characterized by change, the type of institutional matrix determines the option set. For instance, if transaction costs are not negligible (one illustration of markets not being competitive), then the role of institutions is also not negligible, for institutions can also aid the calculation of the expected return on purposive human action, undertaken either individually or through organization, by economizing on transaction costs. Software licenses are the institutions of interest here, the fundamental rules of the organizational game. This brings us to our second, organizational motivation.[5]

Contemporary economic theories of growth – whether neoclassical (Solow 1956; Swan 1956) or endogenous[6] – do not consider production as a set of operations involving intricate knowledge combinatorics and learning. Rather, they consider the organization of inputs as simply the quantities in which the inputs are combined.[7]

---

[4] Buchanan and Yoon (1999) attempt to reconcile the classical, especially Smithian, view of increasing returns with contemporary theories of growth that, especially for distributional reasons, assume constant (or decreasing) returns. Their argument is essentially a technological one: they suggest that it is possible for individual firms to have constant returns to scale production functions and simultaneously have ("generalized") increasing returns throughout the economy. This is so because the constant returns need not be the same for each individual firm. Firms exhibiting different constant returns can define a sort of increasing returns frontier for the economy as a whole (see especially pp. 520-1).

[5] Our two motivations thus embed North's (e.g., 1990, Ch. 1) distinction between institutions as rules of the game and organizations as players of the game.

[6] It can in fact be argued that the neoclassical growth theory and the endogenous one are not so different (Foss 1998).

[7] Two elaborations of this claim are Winter (2006) and Garzarelli (forthcoming).

The neoclassical theory pictures knowledge as an anonymous means to increase output. The endogenous theory sees knowledge as an ordinary good purchased and sold in the market. Consequently, according to the endogenous theory, there is a separation between the producer of the final good and the producer of the knowledge needed for its production. Both theories differ from the Smithian one where the discoverer of new ways of production – the innovator – is the producer himself.[8]

Moreover, if it is true, as just pointed out, that the endogenous theory tries to include innovation which the neoclassical one treats as exogenous, the endogenous still differs from the Smithian theory for trying to model innovation as a rational program. Innovation in Smith is often unplanned[9], not consciously optimized. This is congruent with software production in general, given its complexity (Baetjer 1998). And it is congruent with OSS in specific, where innovation usually emerges thanks to the unpredictable interaction of multiple, often overlapping, inputs.

Our implied analytical method to identify some economic costs and benefits of OSS and proprietary software production directly reflects the nature of our two motivations. This method – known as comparative institutional analysis[10] – does not rest on the more familiar axiomatic optimization-*cum*-equilibrium economic framework, but on an economic framework that, as mentioned, evaluates feasible alternative organizational and institutional arrangements. The axiomatic framework judges the 'efficiency' properties of alternative arrangements in absolute terms: not according to their actual feasibility but according to an unreachable ideal (Arrow 1962). For this reason it should not be too surprising that all feasible alternatives will

---

[8] For example: all ''the improvements in machinery … have by no means been the inventions of those who had occasion to use the machines. Many improvements have been made by the ingenuity of the makers of the machines, when to make them became the business of a peculiar trade; and some by that of those who are called philosophers or men of speculation, whose trade it is not to do anything, but to observe every thing; and who, upon that account, are often capable of combining together the powers of the most distant and dissimilar objects. In the progress of society, philosophy or speculation becomes, like every other employment, the principal or sole trade and occupation of a particular class of citizens'' (Smith 1981[1776], I.i.9, p. 21).

[9] Innovation is also unplanned in most other theories that take technological change seriously, such as the Schumpeterian or neo-Schumpeterian (e.g., Nelson and Winter 1977; Dosi 1982; Langlois and Robertson 1986; Winter 2006). See also Avgerou (1998).

be flawed if compared to such an ideal. The task of comparative institutional analysis is on the other hand to understand which feasible alternative is relatively more promising.[11] The comparison is not among ideal and feasible alternatives, but among feasible alternatives (e.g., firm, market, network, multinational, OSS organization, proprietary software organization) (e.g., Williamson 2005, pp. 11-4). As a result, "there is no guarantee whatsoever that" the "outcome is the most efficient one," such as in, say, the case of a "Walrasian equilibrium or … a Nash equilibrium" (Aoki 2007, p. 10).[12]

The most common criterion to asses the relative 'efficiency' of the alternatives is a negative one: we usually assess the relative ability to internalize negative externalities, such as agency problems, corruption, pollution, rent seeking, resource over-use, shirking, and the like. Seldom do we think that different institutions and organizations exist and persist also – if not principally – to generate positive externalities, most notably rules of the game and division of labor arrangements for both production and exchange of goods, labor, physical and intellectual capital, services, etc. And that it is in the attempt to generate such positive externalities that negative ones may emerge, rather than the other way around (e.g., Garzarelli 2006).

This article focuses on the positive criterion by looking at division of labor dynamics. This is not an unreasonable criterion because, as we shall see, in voluntary OSS production individuals self-select their input: individuals spontaneously align with the problems they are most inclined to solve anyway, leaving little room for unproductive behavior. At the same time, as we shall also see, this does not mean that voluntary production is flawless.

---

[10] See, among an ever-growing literature, the recent Williamson (2005), Greif (2006), and Aoki (2007).

[11] As will become clearer below, this does not mean that comparative institutional analysis does not employ abstraction.

[12] This is not to say that comparative institutional analysis does completely away with any notion of equilibrium. Strictly speaking, the approach is a comparative static one between known alternatives. However, our position in the text is closer to the game-theoretic variant of the exercise. In this variant, the institutions to be compared can themselves be seen as alternative equilibrium systems under common knowledge – where for common knowledge we have in mind the (weaker) definition by Robert J. Aumann as concisely restated by Aoki (2007, pp. 7-8): the "sufficient and necessary" condition "that every agent knows that a" certain "rule is true and that everybody else knows that it is true." See also Greif (2006, pp. 55-152) and Foss and Garzarelli (forthcoming).

To be sure, the method of comparative institutional analysis gives up some of the formal elegance of the more familiar axiomatic framework. However, we believe – and we hope to show – that some loss of formal elegance, which is not tantamount to a loss of rigor, is more than justified in light of the lessons we derive.

## 3. Licenses and Organizational Ideal Types

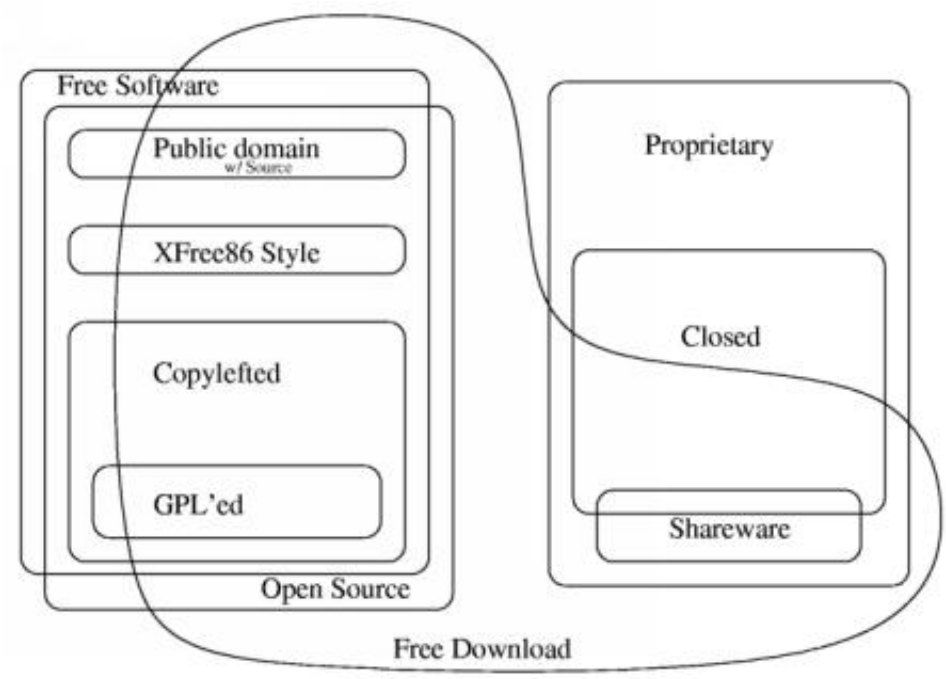
When we think about software we usually think in terms of 'programs,' that is, applications (word processors, spreadsheets, browsers, mail-clients, databases, and so on) that we can run on our computer to get a variety of tasks done. And yet, it must be emphasized that in these cases we use programs only in their executable form. These programs interface with the computer in a machine language, something that the computer is able to read and process. Programs are in fact written in different computer programming languages (BASIC, C, Java, etc.) that are more readable to humans. When programs are in their human-readable form they are called source code. Other types of programs, called interpreters and compilers, are used to translate the source code into its executable form. When programs are in their executable form, all users can use their various functions. But the various functions that can be used are governed by different types of licenses.

If a program is proprietary, the license agreement does not permit anyone to copy, distribute, or modify it. In addition, most of the times the source code is not even available for mere reading or studying. Free Software and OSS programs are instead governed by free software or open source licensing schemes. These licensing schemes specify that the source code is available, can be freely copied, modified, and distributed. Accordingly, the word "free" does not necessarily mean gratis, but the right to access, use and modify the source code. In fact, there are many companies that produce OSS for profit.

For ease of discussion, we place Free Software and Open Source Software under a common rubric of Open Source. And the two software movements do to a large extent share similar rights. But there are differences. One difference is motive (**Wheeler n.d.**). The **Free Software Foundation** (FSF) emphasizes the possibility of

using and sharing software independently from the control of others. This objective is social. The **Open Source Initiative** (OSI) conversely emphasizes a technological objective. It is claimed that programs that fall within the Open Source definition are of higher 'quality.' This 'efficiency' motivation is used to stress the commercial viability of OSS. Yet, in the vast majority of cases, a license that meets the OSI's open source definition also meets the FSF's free software definition. In particular, the most common licenses are both Free and Open: **GPL**, Lesser GPL (**LGPL**), **MIT/X** and **BSD-new** (Wheeler **2005**, **2007**). Figure 1 displays different categories of software.

**Figure 1: Different Categories of Software**




Source: **http://www.gnu.org/philosophy/categories.html**

Different licensing schemes can lead to different organizational forms. The relationship between type of license and organizational form is not always direct or clear cut. The majority of organizational forms exhibit characteristics common to both proprietary and OSS licenses.[13] Nevertheless, we can broadly distinguish two

[13] Cf., e.g., **Eunice (1998)**.

organizational ideal types[14]: cathedral production in the case of proprietary licenses, and bazaar production in the case of OSS licenses (Raymond 2001). In essence, the difference between the two ideal types is that one is a top-down, centralized organization, while the other is a bottom-up, decentralized organization where information is horizontally spread. We can more or less depict the two production ideal types as in Figure 2.[15]

For our purposes, the most important consequence deriving from such organizational distinction is that bazaar organization allows users to participate much more actively in the production process. Users usually tend to participate more because if they have a specific need they can directly tinker with the code in the attempt to satisfy such need, i.e., there is no intermediation. Raymond encapsulates this in terms of a first software programming lesson: "Every good work of software starts by scratching a developer's personal itch" (Raymond 2001, p. 23, emphasis removed). The point to keep in mind is that if the code is modified, then it must be released back to the community. As such, the more active participation that is possible within bazaar OSS projects opens up a cycle of feedback processes, spreading information not just to the community of volunteers but also to the many other users who are not directly involved in the programming. In other words, the rules of the game of bazaar licensing schemes enable the production process to tap from a great variety of inputs – and thus benefit from extensive mistake-ridden learning[16] – because they define an open organization of work.

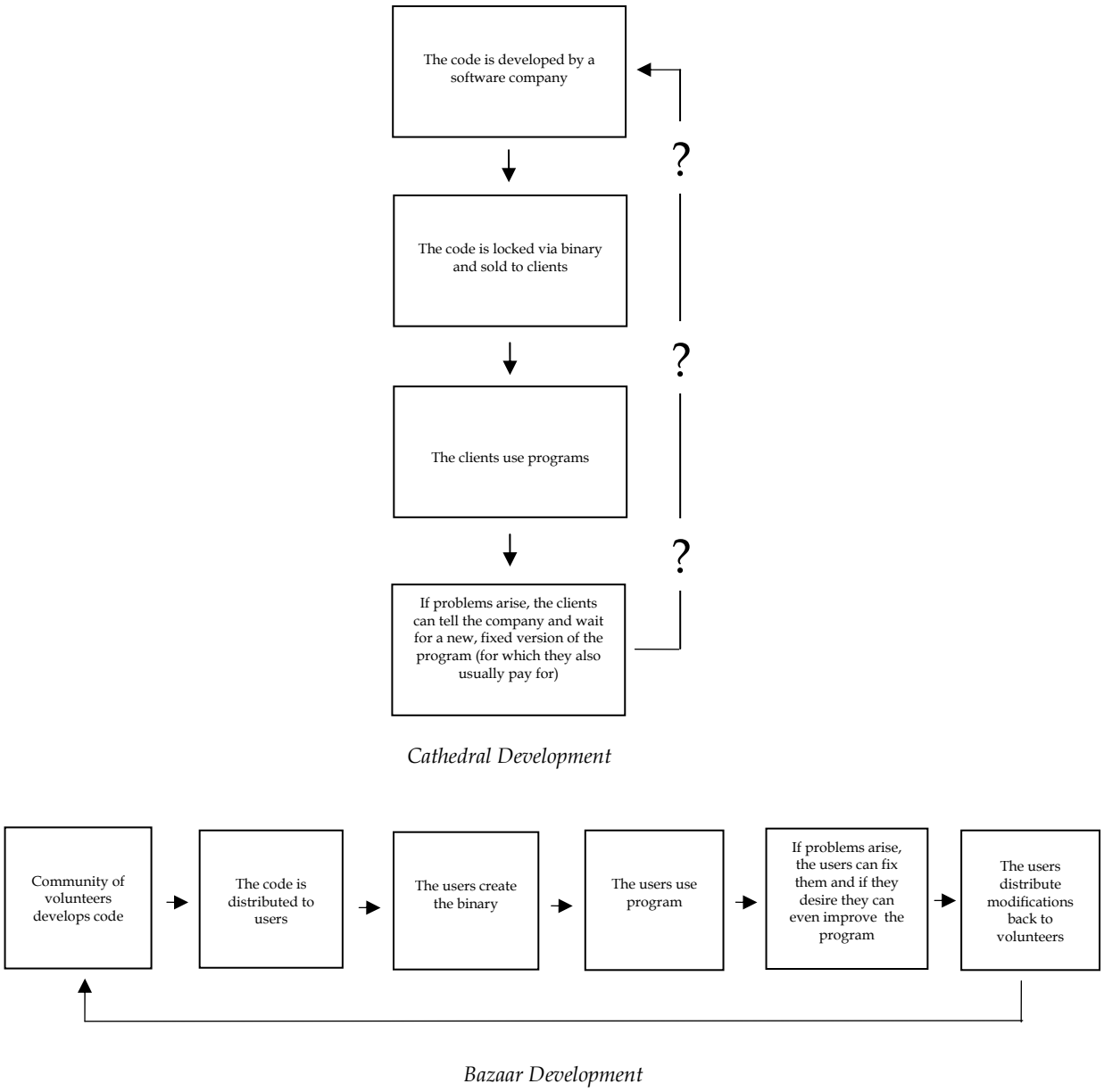
---

[14] Let us stress that we are working with ideal types in the Weberian sense and that as such we are abstracting. An ideal type is a heuristic expedient. It is constructed by considering some characteristics of a certain social role (e.g., the bureaucrat, the entrepreneur, the politician, the professor) or social phenomenon (e.g., the bureau, the firm, the parliament, the university) in order to aid social scientific research. By nature, it is not meant to consider all the characteristics of a role or phenomenon, but some recurring ones. The amount of characteristics an ideal type considers renders it more or less anonymous. In our case, for example, it can be assumed that the cathedral ideal type would – usually – use something like the **waterfall model**. In addition, as will be clarified below, the bazaar ideal type should not be considered as an organizational form where hierarchy or central direction are necessarily absent – Langlois and Garzarelli (2005) present a more formal elaboration of this.

[15] Thanks to Karim Lakhani for allowing us to use a variant of his picture.

[16] On mistake-ridden learning from a theoretical viewpoint, see especially Nelson and Winter (1977); for recent evidence that mistake-ridden learning can be an engine of success, see Anonymous (2007).

**Figure 2: Cathedral versus Bazaar Production**




*Cathedral Development*

*Bazaar Development*

## 4. Two Divisions of Labor

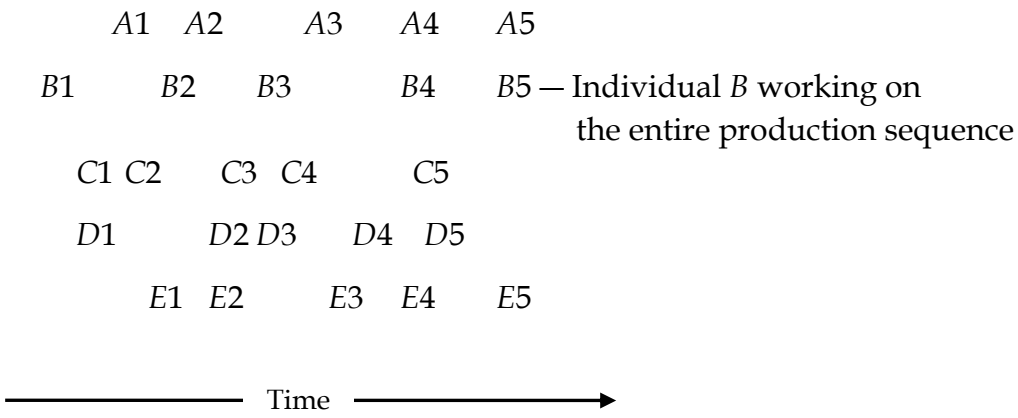
A crucial ingredient of intensive growth is, as Adam Smith pointed out long ago, the division of labor. When the subdivision of tasks is accompanied by market expansion, Smith added, there will be further growth.

The classical division of labor view contains a lot of hidden information. Leijonhufvud (1986) elaborates on such view. Among other important insights,

Leijonhufvud points out that it is possible to identify two types of division of labor: one vertical, the other horizontal.

Imagine five individuals, *A*, *B*, *C*, *D*, *E*, and five production tasks, 1, 2, 3, 4, 5. If the division of labor is such that each of the five individuals performs each of the five tasks sequentially (*A*1, *A*2, *A*3, …) we face a situation of *vertical division of labor*. Think, in general, of crafts production, where the artisan performs every single production task, often times right down to the selling, and the production processes of different artisans are not usually connected. Figure 3 illustrates.

**Figure 3: Vertical Division of Labor**



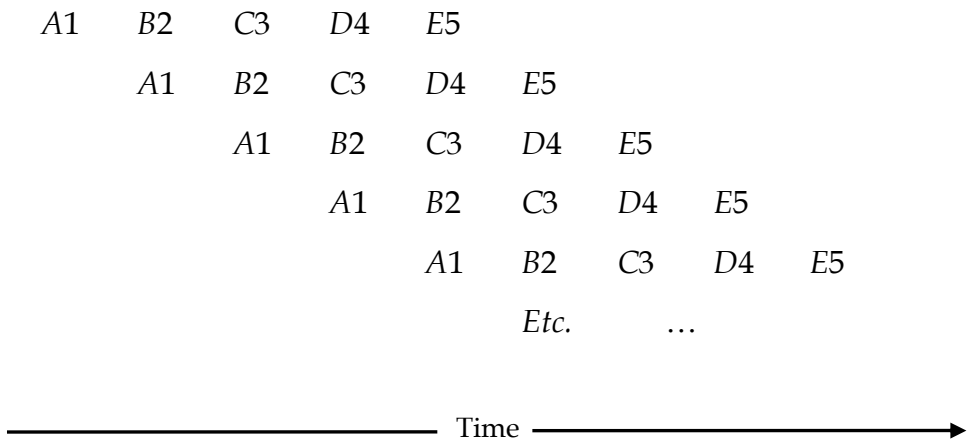

Source: Leijonhufvud (1986, p. 208).

When the market grows to such an extent that it allows each individual to perform only one task, the result is *horizontal division of labor*. In this case, each production task (our 1, 2, 3, 4, 5) becomes a trade in its own. Think, for example, of a shoemaker. Making shoes consists of a number of individual tasks (e.g., cutting the hide, stitching the sole, dying the leather black or brown). If the market grows, then these individual shoemaking tasks can develop into specialized ones. So, in terms of our notation, we would now have individual *A* doing only task 1, individual *B* doing only task 2, individual *C* doing only task 3, etc. Figure 4 depicts the horizontal division of labor.

The two divisions of labor – "the core of [Smith's] theory of production" – slip "through modern production theory as a ghostly technological change coefficient or

as an equally ill-understood economies-of-scale property of the function" (Leijonhufvud 1986, p. 209). But having identified the two divisions is not sufficient for our purposes. We now need to ask how they are comparable.[17]

**Figure 4: Horizontal Division of Labor**




Source: Leijonhufvud (1986, p. 209).

In the case of vertical division of labor we have that each individual is competent, if to different degrees, in a variety of production tasks. The shoemaker is competent in cutting the hide, stitching and dying the leather, making the shoe box for the final sale, and pricing and selling the shoes. There is little specialization at work. What is at work, rather, are both absolute and comparative advantages across the production sequence. This is why it is often commented that artisans are widely skilled or have a wide repertoire of competences in their trade, that their goods are unique, that they have the luxury of being able to work at their own pace, and that their goods are usually more appreciated because unique. Moreover, since the artisan is competent in a variety of tasks he is able to deal more effectively with change across the production sequence, i.e., able to adapt the entire chain of production to an innovation.

But each individual production task, we noted, can become a trade of its own. As a result, in horizontal division of labor there is mostly comparative advantage at

[17] Our answer to this question draws on Leijonhufvud (1986) and Langlois (1988).

work, and this has several benefits. "This great increase of the quantity of work which, in consequence of the division of labor, the same number of people are capable of performing, is owing to three different circumstances; first to the increase of dexterity in every particular workman; secondly, to the saving of the time which is commonly lost in passing from one species of work to another; and lastly, to the invention of a great number of machines which facilitate and abridge labor, and enable one man to do the work of many" (Smith [1776]1981, I.i.5, p. 17). Thus specialization can be an important source of increasing returns, but, under the specific production mode highlighted below, so can be *the lack* of specialization.

Notice, moreover, that these Smithian division of labor economies are not a free lunch. In horizontal division of labor there concurrently is a contraction of the skills of the individual: the competence repertoire of the individual narrows as task efficiency improves.[18] Additionally, output is less unique and more standardized, and the innovation that may occur is less likely to involve more than one stage of production. This implies less ability to adapt to change that is not stage specific and to work at one's own pace since production relies on several stages of production and individuals.

## 5. Extent of the Market and Rules of the Game

The amount of spinning off that turns vertical division of labor into horizontal division of labor primarily thanks to learning effects is bounded by the extent of the market. "As it is the power of exchanging that gives occasion to the division of labor,

[18] This is what the radical literature calls "deskilling" (Marglin 1974): unlike the Marxian view, 'exploitation' here does not derive from machinery but from increasing division of labor. That is, in this view a finer division of labor, not machinery, improves the 'capitalist's' ability to extract surplus from the workers. This view, which originated in the debate about the 'efficiency' of the factory system, however does not consider other mechanisms that operate as the extent of the market grows. If we had only a mechanism in which tasks become simpler and more routine as the extent of the market grows, then such growth would reinforce "deskilling." But if there are, as we try to suggest here, various kinds of confluence and branching of specialization where the growth of relevant knowledge also plays a significant role, then growth in the extent of the market may crowd humans into more skilled occupations. Compare Langlois (2003). See also Babbage (1835, Ch. 20), Rosenberg (1963, 1969), Ames and Rosenberg (1965), and Robertson and Alston (1992) for discussions that even machinery does not necessarily entail "deskilling."

so the extent of this division must always be limited by the extent of that power, or, in other words, by the extent of the market. When the market is very small, no person can have any encouragement to dedicate himself entirely to one employment, for want of the power to exchange all that surplus part of the produce of his own labor, which is over and above his own consumption, for such parts of the produce of other men's labor as he has occasion for" (Smith [1776]1981, I.iii.1, p. 31).

At the outset of the *Wealth of Nations* (I.iii, pp. 31-6) Smith writes that the extent of the market is often limited by three constraints: geographical (presence of natural barriers, such as hills, mountains, and sheer distance), physical (lack of appropriate infrastructure, such as roads), and technological (lack of appropriate means of transport, such as ships). Later, Smith considers the different ability of agricultural and manufacturing societies to cope with a limited extent of the market, and concludes that manufacture requires markets of a larger scale because of its more specialized nature. "Manufactures require a much more extensive market than the most important parts of the rude produce of the land. … Agriculture, therefore, can support itself under the discouragement of a confined market much better than manufactures" (Smith [1776]1981, IV.ix.45, p. 682). In the same context Smith highlights that there may be also another kind of constraint for market expansion: the

> ancient Egyptians had a superstitious aversion to the sea; and as the Gentoo religion [of Indostan] does not permit its followers to light a fire, nor consequently to dress any victuals upon the water, it in effect prohibits them from all distant sea voyages. Both the Egyptians and Indians must have depended almost altogether upon the navigation of other nations for the exportation of their surplus produce; and this dependency, as it must have confined the market, so it must have discouraged the increase of this surplus produce. It must have discouraged, too, the increase of the manufactured produce more than that of the rude produce (Smith [1776]1981, IV.ix.45, p. 682).

In brief, there may be social rules of the game that hold back the diffusion of division of labor.[19] Or, as Greif (2006, p. 56) recently put it in a related context, by "determining who can exchange and what products can be exchanged, … institutions determine the scope and scale of the market."

Another way to capture this institutional observation, more in line with our main concerns, is as follows. If we think of the rules governing OSS and proprietary software we may then say that the OSS ones appear to be more abstract. That is to say that OSS licenses are rules that are more likely to be "applicable to an unknown and indeterminable number of persons and instances," viz., that they "will have to be applied by the individuals in the light of their respective knowledge and purposes" (Hayek [1973]1982, p. 50). In many ways, this is equivalent to saying that the more abstract (or less specific) is an institution, the more likely it is to stimulate the growth of knowledge.

Our reasoning about the costs and benefits of the two divisions of labor and about the extent of the market has hitherto been quite general. It is time to tentatively turn it into some considerations about software organization and economic growth.

## 6. Cathedral, Bazaar, and Division of Labor

What both cathedral and bazaar organization effectively try to do is attempt to tap into the inputs of different individuals. But there are some differences about how the two organizational ideal types do so.

In the cathedral, individuals are hired on the basis of their implied specialization. *A* is hired because he is an expert in requirement analysis (task 1), *B* because he is an expert in specification (task 2), *C* because he is an expert in design and architecture (task 3), *D* because he is an expert in coding (task 4), etc. The organization of work relies on a hierarchy the top constituents of which try to define all the tasks before a project begins. There is a substantial degree of planning so that

[19] A classic exploration of the institutional aspects of Smithian thought is Rosenberg (1960).

everyone is aware not only of the general organizational goal but also of his or her specific goal(s) before any task is begun. The idea is to plan something along the lines of Figure 4: a horizontal division of labor according to expertise in the attempt to simplify a set of complex tasks into smaller sets of less complex tasks in order to achieve a well-defined objective.

The top-down approach to coding typical of the cathedral that attempts to minimize possible production problems ex post by trying to define objectives well ex ante is a classic one. And when the planning is done correctly it is indeed successful: there is no ambiguity in assigned task, deadlines are met, etc. Still, a shortcoming of the cathedral is usually identified. Since all parts of the project are planned in detail, all the teams working on the various parts of the project proceed at the same pace. This entails that it is impossible to begin coding and to check the functioning of the separate parts of the project until the entire project is finished (or almost so). The cathedral organization of work believes, however, that this is more than counterbalanced by the ability to adapt to change that impacts the entire production sequence.

The open rules of bazaar production are instead based on the idea that there should be little conscious planning: the bazaar rests on the assumption that it is impossible to identify – and hence to hire – a set of individuals having the complete knowledge to efficiently solve all possible problems in every point in time. Being based on the assumption of human fallibility due to cognitive limitations[20], the premise is to keep all input options open by not trying to hire the best input ex ante, but by letting individuals self-select their input as situations demand. Tasks are literally taken up by interested programmers as they emerge. Thus, bazaar division of labor takes advantage of effort in a spontaneous fashion (Langlois and Garzarelli 2005). In this way, it is able to rely on a large – in effect, potentially unlimited – knowledge pool. While "coding remains an essentially solitary activity, the really great hacks come from harnessing the attention and brainpower of entire

[20] The *locus classicus* of this argument is Hayek (1945).

communities. The developer who uses only his or her own brain in a closed project is going to fall behind the developer who knows how to create an open, evolutionary context in which feedback exploring the design space, code contributions, bug-spotting, and other improvements come from hundreds (perhaps thousands) of people" (Raymond 2001, pp. 50-1).

Notice that to tap from effort in a spontaneous way does not necessarily imply a horizontal division of labor in the traditional sense. There may be thousands of individuals each working in parallel on the same task. This means that, unlike the cathedral ideal type, it is possible to code and test early on. In fact, in the bazaar the details, e.g., the connections among various parts of the program, are left for last. This organization of work consequently exhibits the ability to adapt to change of a stage-specific type.

But the fact that several individuals may be working on multiple tasks at once also more generally means that bazaar organization can simultaneously present both vertical and horizontal division of labor characteristics, implying that there is some redundancy at play. The redundancy is not a shortcoming, however. It engenders economies that have the ability to capitalize on multiple, intersecting knowledge combinatorics.[21] These *redundant economies* from parallel, overlapping inputs encourage a production mode whereby being specialized in a particular task is not a *condicio sine qua non* to contribute: a contributor may still be a programmer in the formal sense or a final user with programming skills, but his input(s) may not always directly reflect his primary specialty. What matters is the spontaneity of the contribution, because the shared belief is that there's potentially something to learn from everyone. In fact, as we noted, the central tenet of the bazaar is that the benefits of tapping from a virtually unlimited knowledge pool through a redundant division of labor outweigh all other costs, including coordination costs.

---

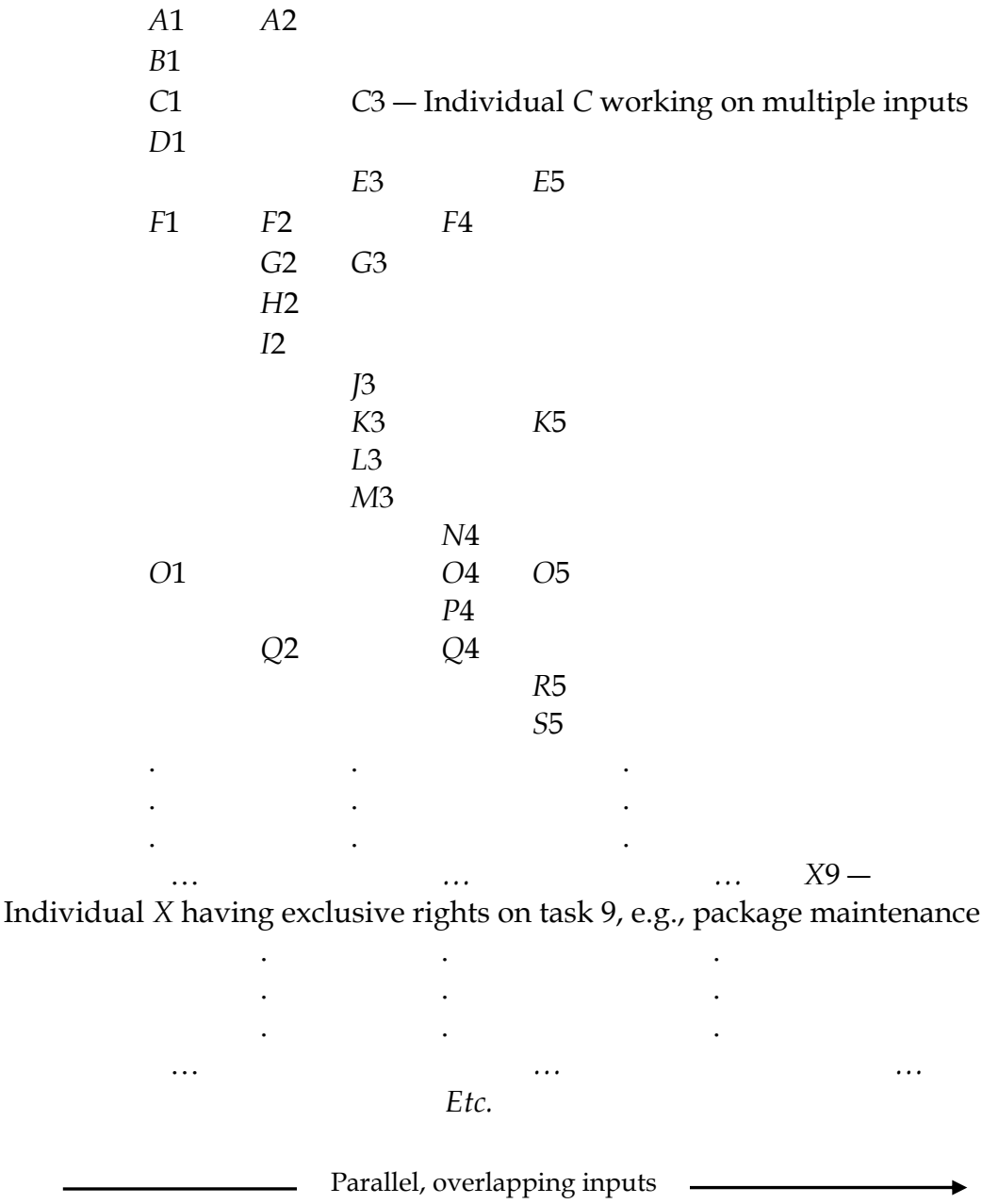
[21] These redundant economies have similar effects to – even though they are not fully isomorphic to – what Garud and Kumaraswamy (1995, p. 96) christen economies of substitution, "which exist when the cost of designing a higher-performance system through the partial retention of existing components is lower than the cost of designing the system afresh."

Redundant economies entail that – somewhat paradoxically – there can be increasing returns not just from specialized spontaneous inputs but also from nonspecialized spontaneous ones (in the sense just defined). This holds to the extent that the spillovers are nonrivalrous, namely, to the extent that the appropriation of the spillovers by one user does not affect the appropriation on behalf of other users (Garzarelli 2004). As already mentioned, this has empirical corroboration, for not all OSS bazaar projects are started from scratch, but often build on previous code (Haefliger et al. forthcoming): there is substantial knowledge reuse and sharing as well as knowledge discovery from experimentation without congesting appropriation. Redundant economies create value as long as individuals

> can plan the use of their resources free of interference from unpredictable external influences. This enables [planning and acting] despite the limited, local nature of most knowledge; it thus permits more effective use of divided knowledge, aiding the division of labor. The value of protected spheres and local knowledge has thus far been the sole motivation for giving software modules 'property rights' through encapsulation. … Encapsulation and communication of resources correspond to ownership and voluntary transfer, the basis of trade. … [M]otivated by the need for decentralized planning and division of labor, computer science has reinvented the notion of property rights.[22]

[22] **Miller and Drexler (1988, webbed version)**. In his classic contribution, Alchian (1977[1965], p. 140, original emphasis) highlighted the following benefits of well-defined property rights: "(1) concentration of rewards and costs *more* directly on each person responsible for them, and (2) comparative advantage effects of specialized applications of (a) knowledge in control and (b) of risk bearing." This argument is elaborated in Langlois and Garzarelli (2005) in the specific context of modularity and OSS organization; compare also Simon ([1962]1998, 2002) on the notion of near-decomposability; Conway (1968) on the idea that output of production reflects the process of production; Sanchez and Mahoney (1996), which claims that product designs organization; and Raymond (2001, pp. 65-111) for a Lockean property right theory take on OSS.

**Figure 5: Bazaar Division of Labor**




In terms of our notation, in bazaar production we may have cases like *A*1, *B*1, *C*1, *D*1, *F*1, *O*1; *A*2, *F*2, *G*2, *H*2, *I*2, *Q*2; *C*3, *E*3, *G*3, *J*3, *K*3, *L*3, *M*3; etc. Figure 5 tries to give an idea of this redundant division of labor with both horizontal and vertical characteristics within an individual project. Let us further note that in bazaar projects it is not impossible to have one individual having (de jure) exclusive rights on a single package (a specific software component or set of software components). One concrete illustration of this is the case of the **Package Maintainers** of the **Debian**

**Project**. This is why *X*9 in Figure 5 is essentially an isolated case: no one else is working on task 9 because it is not allowed. Contemporaneously, this does not mean that other de facto isolated cases are necessarily absent in any point in time. We may easily have the case where only one individual is interested in working on one task.

Before moving on, let us additionally qualify that we do not want to be read as claiming that the bazaar production mode is unique. Similar modes include the professions (Garzarelli 2004), "open science" (Dalle and David 2004), online open bibliographic databases, such as Research Papers in Economics (**RePEc**) (Krichel and Zimmermann 2005), and collaboration for literary and hobbyist ends (like the online encyclopedia **Wikipedia** and the photography site **photo.net**).

Embedded in our discussion is the other side of the equation, too: the extent of the market. In our stylized illustration the rules of bazaar production place virtually no limit on the number of individuals working on a particular aspect of a software project. Even though not all individual inputs may ultimately end up in a program – that is, even if not all of the work in parallel is ultimately used in the particular software for which it is aimed (as, indeed, is often the case) – the extent of the market is still growing. The extent of the market is growing because mistake-ridden learning is nonetheless taking place.[23] And it is learning that, as we saw, leads to some of the horizontal benefits that Smith was so keen on: improvements in individual skills and the ability to focus on a particular task, which would lead to the recognition of new organizational and technological opportunities.[24] This would give rise, moreover, to new individuals participating in the production process and to the birth of new software projects that draw on the knowledge pool of other, preceding projects.

Take note that the other benefit of horizontal division of labor – the reduction of idle time between tasks or reduction of "sauntering" (Smith [1776]1981, I.1.7, pp. 18-9) – is really immaterial in the bazaar. Indeed, sauntering is actually a main raison

[23] Let us qualify at this juncture that the mistake-ridden learning that can create market expansion may not only be caused by the bazaar. There is arguably a cross-fertilization effect in that often many programmers work on both cathedral (by day) and bazaar (by night) projects. Thanks to the anonymous referee for reminding us about this.

d'être of the bazaar: everyone is able to work at his or her pace like in the vertical division of labor. The reason why the reduction of sauntering is immaterial is due to what the cathedral considers inefficient: the economies from redundancy. Sauntering is offset by the redundant economies of the bazaar, that is, it is more than compensated by the benefits of networking in the first place. In "the open-source community organizational form and function match on many levels. The network is everything and everywhere: not just the Internet, but the people doing the work form a distributed, loosely coupled, peer-to-peer network that provides multiple redundancy and degrades very gracefully. In both networks," i.e., the Internet and the bazaar, "each node is important only to the extent that other nodes want to cooperate with it" (Raymond 2001, p. 224, note 10).

This is not to say that problems arising from sauntering would be completely absent. As a bazaar projects grows, so does the possibility that individuals lose track of priorities. In large projects (e.g., Debian) package maintainers usually have, as was said earlier, complete authority over their part of the project. But because they are volunteers, it is not uncommon for them to lose interest or become busy with something else that is not, e.g., a top priority. If they stop maintaining their package – one possible nonroutine pattern of behavior that could also lead to lock-in[25] – the quality of the overall project may suffer over time. It is in these cases that the hierarchy more typical of the cathedral manifests itself: the leader of the project intervenes. A bazaar project leader is at the top of the organizational hierarchy. Traditionally, a leader has authority to grant some rights to developers and is expected to intervene in times of "urgent action." It would seem that in actual fact the most fundamental role that he or she plays is one that has mostly to do with proactive and suggestive action; a leader is in essence expected to coordinate rather

---

[24] Compare Rosenberg (1969), Nelson and Winter (1977), and Dosi (1982).

[25] Following David (1985), by lock-in (or path dependency) economists have in mind the notion that a certain institution, organization or technology gets stuck in a particular trajectory that, with ex post wisdom, may not seem the most appropriate one. That is, the notion is that a particular standard may reveal itself to be 'inefficient'. David's controversial (cf. Liebowitz and Margolis 1990) example is the QWERTY keyboard, claimed to be inferior to the (nonstandard) one of Dvorak. A switching out of such 'lock-in' is very unlikely because it would involve a fixed cost for many individuals. So the proposal is that, if caught in time, that is, before it becomes a standard, a particular lock-in can be 'cured' by fiat.

than to direct production.[26] This is so because the leadership role itself is a fruit of openness. As Linus Torvalds, creator and principal software architect of Linux, says: it

> just happened by a kind of natural selection. I'd been doing Linux as my own personal project, and I put it out just because I wanted comments and because I thought that somebody else was interested, and obviously, partly because I thought it was a really interesting project and it's a way of just showing off … there were a lot of things that people asked for and also implemented themselves. They started out just asking for small things and then asking for larger things or doing them themselves, and none of this was very planned for. The leadership part came by default, because nobody wants to make decisions, right? Things just happened, and it wasn't really planned. And I was the obvious person for it.[27]

## 7. Conclusions

The division of labor relationships at interest here are not so straightforward. They are, rather, redundant, combining vertical and horizontal characteristics: everyone is typically specialized in the realization of one task that he or she volunteers input for, but, thanks to open organizational rules, this specialization does not prevent individuals from trying to contribute to other tasks where task matching may be weaker. This results in a simultaneous increase in the productivity of all tasks, because total output increases at a rate superior to the increase in the number of programmers (or hardware).

---

[26] See, e.g., **http://www.debian.org/devel/leader.en.html**

[27] March 17, 2000, **http://www.taborcommunications.com/archives/17276.html**

In addition to productivity gains of the more traditional sort, bazaar OSS production can foster growth in the extent of the market through the faster appropriability of the benefits from innovation. One of the distinguishing marks of OSS, we noted, is that its users have the freedom not only to study its code but also to modify it. In other words, the production process is open not just to the programmer in the more formal sense, but to the final user as well. If a final user has some programming skills, he can try to satisfy a particular software need by directly accessing the code. Hence, if the final user successfully modifies a program, this modification is in turn accessible to others. And even if a final user does not have sufficient programming skills to satisfy a particular need, he can in principle improve them thanks to the freedom of access to the code. Productivity improves even if not all inputs are ultimately accepted for a project, for there are still benefits from concomitant volunteer code submission (e.g., improvement of skill, knowledge diffusion, reuse of code, trial and error learning). One substantive implication is that the speed at which innovations become available to users is faster than under a proprietary software regime, which implies even more innovations – and therefore more growth.

The ability of bazaar organization to create value is not only restricted to the gains of productivity and to diffused innovation within a single bazaar project or within the more general OSS community. Bazaar OSS can play a crucial role at a larger level too. It can contribute to narrowing the technological gap between developing and developed countries, and therefore decreasing the so-called dependency of developing countries on foreign technology. Additionally, OSS presents an opportunity for developing countries to customize technology by allowing individuals to adapt the existing technology to their specific needs and conditions, which are not necessarily the same as those in developed countries.[28]

Similarly, by allowing developing countries to participate in technology development, OSS becomes a crucial factor for these countries to develop their own

[28] Compare, for example, Arora et al. (1997, p. 133).

industries (including for profit OSS firms). We must not forget, in fact, that it is possible for production processes to split. The output of one process can become the input for another, and so on. An economy experiencing growth will present increasing division of labor not just within organizations and sectors, but also among organizations and sectors.[29] If history is anything to go by, an economy that is self-sustaining in the long term is one that experiences industrialization and learning concurrently.[30]

Yet, this article does not wish to claim a mechanical relationship between software and growth as is sometimes done in the case of hardware.[31] The article's approach is more microanalytic. By comparing the costs and benefits of proprietary and OSS organization of work – the cathedral and the bazaar organizational ideal types – it has argued that, other things equal, the bazaar is more promising for growth because of the possibility to spontaneously participate to a software project.[32] In short, our supply-side perspective leads us to conjecture that the bazaar has a value-stimulating edge because, thanks to its licenses, the costs of drawing on both vertical and horizontal division of labor are more than counterbalanced by the benefits, which include growth in the extent of the market.

At the same time, however, we do not want to be considered as contemporary reincarnations of Voltaire's **Dr. Pangloss**. OSS and bazaar organization are not the best of all possible worlds. As with most other technologies whose purposes go beyond their original scope – think of computers, machine tools, railroads, semiconductors, the steam engine, transistors, etc.[33] – a minimum of literacy, competence and infrastructure is required.

---

[29] For example, Young (1928), Leijonhufvud (1986), and Langlois (1988).

[30] E.g., Rosenberg (1963).

[31] For notable exceptions, see, for example Ames and Rosenberg (1965), Robertson and Alston (1992), and Avgerou (1998).

[32] Admittedly, a great part of our "other things equal" has to do with exploring the bazaar more than the cathedral, that is, with keeping the cathedral more an (anonymous) ideal type than the bazaar.

[33] In economics, these are known as general-purpose technologies – see, for example, Bresnahan and Gambardella (1998), and Rosenberg's (1963) earlier notion of "technological convergence"; see also Stigler (1951).

If the freedom of OSS licenses is more fully understood and tapped into, we would perhaps in part accomplish what growth and development policies have always searched for: ways of transferring relevant productive knowledge and of creating sustainable growth through the generation of more productive knowledge. There is more to OSS than the fact that some OSS programs are available gratis. OSS is above all a system of rules that aids the spontaneous "release of energy."[34]

**Acknowledgements.** A preliminary draft of this article was presented at the World Forum on Information Society, "Digital Divide, Global Development, and the Information Society," November 14-16, 2005, a parallel event of the Second Phase of the World Summit on the Information Society, Tunis, Tunisia. We wish to thank the participants at the event for feedback, and especially acknowledge the very helpful comments of an anonymous referee and Editor of this journal. We also thank Roberto Galoppini, Lyndal Keeton, Richard Langlois, the late Bubele Mahanjana, Martin Michlmayr, and Robert Vivian. The first and third authors also acknowledge financial support from the American University of Rome. The first author moreover acknowledges the Centro di Metodologia delle Scienze Sociali at LUISS-Guido Carli, Rome, particularly Dario Antiseri and Caterina Galluccio, for hospitality from November 2006 to January 2007.

## References

Alchian, Armen A. 1977. "Some Economics of Property Rights," in *Idem*, *Economic Forces at Work*. Indianapolis: Liberty Fund, Inc.: 127-149. Originally published in 1965, *Il Politico* 30(4): 816-829.

Ames, Edward, and Nathan Rosenberg 1965. "The Progressive Division and Specialization of Industries," *Journal of Development Studies* 1(4): 363–383(July).

Anonymous 2007. "**Lessons from Apple: What Other Companies Can Learn from California's Master of Innovation**," *The Economist* 383(8532): 11(June 9-15).

Aoki, Masahiko 2007. "**Endogenizing Institutions and Institutional Changes**," *Journal of Institutional Economics* 3(1): 1-31(April).

Arora, Ashish, Alfonso Gambardella and Enzo Rullani 1997. "Division of Labor and the Locus of Inventive Activity," *Journal of Management and Governance* 1(1): 123-140(November).

Arrow, Kenneth J. 1962. "Economic Welfare and the Allocation of Resources to Invention," in Richard R. Nelson, ed., *The Rate and Direction of Inventive Activity: Economic and Social Factors*. Princeton: Princeton University Press: 609-625.

[34] An expression borrowed from Hurst (1967).

Avgerou, Chrisanthi 1998. "How Can IT Enable Economic Growth in Developing Countries?," *Information Technology for Development* 8(1): 15-28(January).

Babbage, Charles 1835. ***On the Economy of Machinery and Manufactures***, 4th edition. London: Charles Knight.

Baetjer, Howard Jr. 1998. *Software as Capital. An Economic Perspective on Software Engineering*. Los Alamitos, CA: IEEE Computer Society.

Benkler, Yochai 2002. "**Coase's Penguin, or, Linux and the Nature of the Firm**," *Yale Law Journal* 112(3): 369-446(December).

Bresnahan, Timothy F. and Alfonso Gambardella 1998. "**The Division of Inventive Labor and the Extent of the Market**," in Elhanan Helpman, ed., *General Purpose Technologies and Economic Growth*. Cambridge, Mass.: MIT Press: 253-281.

Buchanan, James M., and Yong J. Yoon 1999 "**Generalized Increasing Returns, Euler's Theorem, and Competitive Equilibrium**," *History of Political Economy* 31(3): 511-523(Fall).

Conway, Melvin E. 1968. "**How do Committees Invent?**," *Datamation* 14(4): 28-31(April).

Dalle, Jean-Michelle and Paul A. David 2004. "**The Allocation of Software Development Resources in Open Source Production Mode**," in Joseph Feller, Brian Fitzgerald, Scott A. Hissam and Karim Lakhani, eds., *Perspectives on Free and Open Source Software*. Cambridge, MA: MIT Press: 297-328.

David, Paul A. 1985. "**Clio and the Economics of QWERTY**," *American Economic Review, Papers and Proceedings* 75(2): 332-337(May).

Dosi, Giovanni 1982. "Technological Paradigms and Technological Trajectories: A Suggested Interpretation of the Determinants and Directions of Technical Change," *Research Policy* 11(3): 147-162(June).

Eunice, Jonathan 1998. *Beyond the Cathedral, Beyond the Bazaar* (May 11). Online: **http://www.illuminata.com/cgi-local/pub.cgi?docid=cathedral**

Foss, Nicolai J. 1998. "**The New Growth Theory: Some Intellectual Growth Accounting**," *Journal of Economic Methodology* 5(2): 223-246(December).

Foss, Nicolai J. and Giampaolo Garzarelli forthcoming. "**Institutions as Knowledge Capital: Ludwig M. Lachmann's Interpretative Institutionalism**," *Cambridge Journal of Economics*.

Garud, Raghu, and Arun Kumaraswamy 1995. "**Technological and Organizational Designs for Realizing Economies of Substitution**," *Strategic Management Journal* 16(Special Issue: Technological Transformation and the New Competitive Landscape): 93-109(Summer).

Garzarelli, Giampaolo 2004. "**Open Source Software and the Economics of Organization**," in Jack Birner and Pierre Garrouste, eds., *Markets, Information and Communication*. London and New York: Routledge: 47-62.

Garzarelli, Giampaolo 2006. "**Cognition, Incentives, and Public Governance: Laboratory Federalism from the Organizational Viewpoint**," *Public Finance Review* 34(3): 235-257(May).

Garzarelli, Giampaolo forthcoming. "The Organizational Approach of Capability Theory: A Review Essay," *Review of Political Economy.*

Greif, Avner 2006. *Institutions and the Path to the Modern Economy: Lessons from Medieval Trade.* Cambridge: Cambridge University Press.

Haefliger, Stefan, Georg von Krogh and Sebastian Spaeth forthcoming. "Code Reuse in Open Source Software," *Management Science.*

Hayek, Friedrich A. von 1945. "**The Use of Knowledge in Society**," *American Economic Review* 35(4): 519-530(September).

Hayek, Friedrich A. von 1982. *Law, Legislation and Liberty. Volume 1: Rules and Order* (First published in one volume with corrections and revised preface in 1982). London: Routledge. Volume 1 first published 1973.

Hurst, James Willard 1967. *Law and the Conditions of Freedom in the Nineteenth-century United States*. Madison, Milwaukee, and London: The University of Wisconsin Press.

Krichel, Thomas, and Christian Zimmerman 2005. "The Economics of Open Bibliographic Data Provision," Working Paper 2005-01, University of Connecticuct, Storrs, Department of Economics. Online: **http://ideas.repec.org/p/uct/uconnp/2005-01.html**

Lakhani, Karim R. and Eric von Hippel 2003. "**How Open Source Software Works: 'Free' User-to-User Assistance**," *Research Policy* 32(6): 923-943(June).

Langlois, Richard N. 1988. "Economic Change and the Boundaries of the Firm," *Journal of Institutional and Theoretical Economics* 144(4): 635-657(September).

Langlois, Richard N. 1999. "**Scale, Scope, and the Reuse of Knowledge**," in Sheila C. Dow and Peter E. Earl, eds., *Economic Organization and Economic Knowledge: Essays in Honor of Brian J. Loasby.* Cheltenham: Edward Elgar: 239-254.

Langlois, Richard N. 2003. "**Cognitive Comparative Advantage and the Organization of Work: Lesson from Herbert Simon's Vision of the Future**," *Journal of Economic Psychology* 24(2): 167-187(April).

Langlois, Richard N., and Giampaolo Garzarelli 2005. "Of Hackers and Hairdressers: Modularity and the Organizational Economics of Open-source Collaboration," Unpublished Paper.

Langlois, Richard N. and Paul L. Robertson 1996. "**Stop Crying over Spilt Knowledge: A Critical Look at the Theory of Spillovers and Technical Change**," Paper presented at the MERIT conference on Innovation, Evolution and Technology, August 25-27, Maastricht, the Netherlands.

Leijonhufvud, Axel 1986. "**Capitalism and the Factory System**," in Richard N. Langlois, ed., *Economics as a Process: Essays in the New Institutional Economics*. New York: Cambridge University Press: 203-223.

Lerner, Josh, and Jean Tirole 2002. "Some Simple Economics of Open-source," *Journal of Industrial Economics* 50(2): 197-234(June).

Liebowitz, S. J., and Stephen E. Margolis 1990. "**The Fable of the Keys**," *Journal of Law and Economics* 33(1): 1-25(April).

Lucas, Robert E. 1988. "On the Mechanics of Economic Development," *Journal of Monetary Economics* 22(1): 3-42(July).

Marglin, Stephen A. 1974. "**What Do Bosses Do? The Origin and Functions of Hierarchy in Capitalist Production**," *Review of Radical Political Economics* 6(2): 60-112(Summer).

Miller, Mark S., and K. Eric Drexler 1988. "**Markets and Computation: Agoric Open Systems**," in Bernando A. Huberman, ed., *The Ecology of Computation*. Amsterdam: North-Holland: 133-176.

Nelson, Richard R., and Sidney G. Winter 1977. "In Search of Useful Theory of Innovation," *Research Policy* 6(1): 36-76(January).

North, Douglass C. 1990. *Institutions, Institutional Change and Economic Performance*. New York: Cambridge University Press.

Raymond, Eric S. 2001. ***The Cathedral and the Bazaar. Musings on Linux and Open Source by an Accidental Revolutionary*** (revised edition). Sebastopol, CA: O'Reilly & Associates, Inc.

Robertson, Paul L. and Lee J. Alston 1992. "Technological Choice and the Organization of Work in Capitalist Firms," *Economic History Review* (New Series) 45(2): 330-349(May).

Romer, Paul M. 1986. "Increasing Returns and Long-Run Growth," *Journal of Political Economy* 94(5): 1002-1037(October).

Rosenberg, Nathan 1960. "**Some Institutional Aspects of the Wealth of Nations**," *Journal of Political Economy* 68(6): 557-570(December).

Rosenberg, Nathan 1963. "Technological Change in the Machine Tool Industry, 1840-1910," *Journal of Economic History* 23(4): 414-443(December).

Rosenberg, Nathan 1969. "The Direction of Technological Change: Inducement Mechanisms and Focusing Devices," *Economic Development and Cultural Change* 18(1), Part I: 1-24(October).

Sanchez, Ron, and Joseph T. Mahoney 1996. "Modularity, Flexibility, and Knowledge Management in Product and Organizational Design," *Strategic Management Journal* 17(Special issue: Knowledge and the Firm): 63–76(Winter).

Simon, Herbert A. 1998. "The Architecture of Complexity: Hierarchic Systems," in *Idem*, *The Sciences of the Artificial*, 3rd edition, second printing. Cambridge, Mass.: MIT Press: 183-216. Originally published in 1962, *Proceedings of the American Philosophical Society* 106(6): 467-482(December 12).

Simon, Herbert A. 2002. "Near Decomposability and the Speed of Evolution," *Industrial and Corporate Change* 11(3): 587-599(June).

Smith, Adam 1981. ***An Inquiry into the Nature and Causes of the Wealth of Nations*** (two volumes). Indianapolis: Liberty Fund. First published 1776.

Solow, Robert M. 1956. "**A Contribution to the Theory of Economic Growth**," *Quarterly Journal of Economics* 70(1):65-94(February).

Stigler, George J. 1951. "**The Division of Labor Is Limited by the Extent of the Market**," *Journal of Political Economy* 59(3): 185-193(June).

Swan, Trevor W. 1956. "Economic Growth and Capital Accumulation," *Economic Record* 32(2): 334-361.

UNCTAD 2003. ***E-Commerce and Development Report 2003***. New York and Geneva: United Nations.

Wheeler, David A. n.d. "Open-source Software/Free Software (OSS/FS) References." Online: **http://www.dwheeler.com/oss_fs_refs.html**

Wheeler, David A. 2007. "Make your Open-source Software GPL-Compatible. Or Else." (March 7) Online: **http://www.dwheeler.com/essays/gpl-compatible.html**

Wheeler, David A. 2005. "Why Open-source Software/Free Software (OSS/FS)? Look at the Numbers!" (November 14). Online: **http://www.dwheeler.com/oss_fs_why.html**

Williamson, Oliver E. 2005. "**The Economics of Governance**," *American Economic Review, Papers and Proceedings* 95(2): 1-18(May).

Winter, Sidney G. 2006. "**Toward a neo-Schumpeterian Theory of the Firm**," *Industrial and Corporate Change* 16(1):125-141(February).

Young, Allyn A. 1928. "**Increasing Returns and Economic Progress**," *Economic Journal* 38(152): 527-542(December).

Young, Allyn A. 1990. "Big Business: How the Economic System Grows and Evolves like a Living Organism. What Will Become of the Small Retailer?," in Roger J. Sandilands, ed., *Journal of Economic Studies*, 17(3/4: Special Issue on *Nicholas Kaldor's Notes on Allyn Young's LSE Lectures, 1927-29*): 161-170. Originally published anonymously in 1929, *The Book of Popular Science*, Vol. 15, New York, Ch. 38.